Assessment activity: Design project

ENGN4339: Semester 2, 2025

School of Engineering, College of Systems and Society

Kawsihen Elankumaran / Research Fellow / SoEN / CSS / ANU

Docking at the International Space Station

This document presents the description, requirements, and assessment criteria for the design project activity. Students are also encouraged to attend a briefing session in Week 2 to obtain additional supportive information. The design project is a group activity with a maximum of 3 members per group. The design project aligns with the Learning Outcome 3 and 6 on the formulation, computational simulation and performance evaluation of an integrated aerospace system. Any changes to the description/requirements of the activity will be updated via *Wattle*.

In this design project, we will formulate and design a method for the docking of a small cargo resupply spacecraft (hereafter referred to as *Ceres*) at the International Space Station (ISS). This activity helps students apply their understanding of fundamental concepts in orbital mechanics to develop comprehensive solutions to a practical design problem. Engagement in design projects requires critical thinking, creativity, and knowledge acquisition. An in-depth knowledge of orbital mechanics, however, is not required to get started on this project and most of the essential information is outlined in this document. Week 2 lectures will provide a formal introduction to orbital mechanics and rendezvous.

1. Background

The International Space Station (ISS) is a crewed space platform operating in low-Earth orbit administered by intergovernmental agreements among participating countries. The ISS serves to advance knowledge and capability related to space sciences and engineering. It serves as a hub for performing scientific research (on astronomy and astrobiology for example), and as a platform for experimentation, testing and qualification of space hardware. The ISS is regularly resupplied by space vehicles that transport essential life supplies and experimental hardware. To access the ISS, the incoming space vehicle must dock or berth at one of its ports. This allows the hatches to be opened, and the vehicle accessed.

2. Motivation

The following references help provide additional hands-on experience on the ISS operations and docking process. These resources are optional and not essential to undertake the project.

- Take a virtual walk inside the ISS and check out the various modules and their functions.

  https://artsandculture.google.com/streetview/international-space-station/WgFE9b04h8A0ww

- Check out the virtual ISS docking simulator online. Note that this simulator includes attitude maneuvers such as pitch, roll and yaw which we will not consider in this design project for simplicity. We will also not design a visual graphical simulator and use simple 3D trajectory plots for visualization.

  https://iss-sim.spacex.com/

3. Technical description

From a flight mechanics perspective, docking can be best understood from the target-relative motion of spacecraft. In this regard, docking is achieved when the position and velocity of an incoming spacecraft relative to the ISS is sufficiently small. Note that docking also requires the spacecraft to be correctly orientated to the docking port in the ISS (i.e., attitude error to be sufficiently small), but we will disregard this requirement for the sake of simplicity of this project.

Satellites and space vehicles operating around the Earth (or a central body) remain in an orbit. In flight mechanics, Clohessy-Wiltshire (CW) equations express the relative motion of a chaser spacecraft in a circular or elliptical orbit, relative to a target spacecraft in a circular orbit. The gravitational influence of the central body is considered as a point mass. The CW equations are in fact a first-order approximation. The motion is expressed in a coordinate frame centered at the target spacecraft. In this setting, the CW equations of motion assume the following form for a chaser spacecraft whose state is represented by $[x\ y\ z\ \dot{x}\ \dot{y}\ \dot{z}]^{\mathrm{T}}$.

$$\begin{aligned} \ddot{x} &= 3n^2x + 2n\dot{y} \\ \ddot{y} &= -2n\dot{x} \\ \ddot{z} &= -n^2z \end{aligned} \qquad \text{(Eq. 1)}$$

Regarding the axes of the aforementioned coordinate frame, x is measured radially outward from the target spacecraft, while y is in the orbit track direction of the target spacecraft. *z* is measured along the orbital angular momentum vector of the target spacecraft and n represents its orbital rate given by $\sqrt{\frac{\mu}{a^3}}$, where a is the radius of the orbit (semi-major axis) of the target spacecraft around the central body and μ is the gravitational parameter of the central body (Please see the diagram in the Discussion session slides).

In our docking problem, the target is the ISS, the chaser is *Ceres* and the central body is the Earth. We will further explore the concepts related to space rendezvous in detail in Lecture 2. Students are also encouraged to seek additional knowledge from online resources about CW equations.

The expressions in Eq. 1 refer to a ballistic (i.e. gravitational) motion of *Ceres*. However, to perform docking, *Ceres* requires propulsion (thrust acceleration) to actively maneuver itself towards the ISS. In this case, the equations of motion can be rewritten in the following manner (in the same CW frame) considering the accelerations *u* generated by its onboard thrusters.

$$\begin{aligned} \ddot{x} &= 3n^2x + 2n\dot{y} + u_x \\ \ddot{y} &= -2n\dot{x} + u_y \\ \ddot{z} &= -n^2 z + u_z \end{aligned} \qquad \text{(Eq. 2)}$$

Students may recall from Lecture 1 that the design of space systems must consider onboard resource limitations. In this regard, maneuvers for docking must be designed in a manner that minimizes the total propulsion requirement for *Ceres*. To this end, the corresponding Performance Index to be minimized between time t_i and t_f is given by *J* as follows.

$$J = \frac{1}{2}\int_{t_i}^{t_f} U^T U \, dt$$

where

$$U = [u_x \; u_y \; u_z]^T$$

4. Objective

The objective of this design project is to design an input acceleration profile that would eventually accomplish the docking of *Ceres* at ISS. The input acceleration profile refers to how the applied acceleration u varies with time (this applied acceleration can be interpreted as the control input). The overall goal is to dock *Ceres* at ISS with the minimum input requirement, expressed by Performance Index J.

The following plots are required at the completion of the project.

- Plot 1: Plot of applied acceleration u_x, u_y and u_z vs time.
- Plot 2: Plot of distance (magnitude) between *Ceres* and ISS vs time.
- Plot 3: Plot of the magnitude of ISS-relative velocity of *Ceres* vs time.
- Plot 4: Plot of performance index J vs time.

Visualization of the target relative trajectory of *Ceres* in the form of a 3D plot is also required for verification. Note that the ISS would always be at the origin of this plot, due to the target relative representation of motion by CW equations.

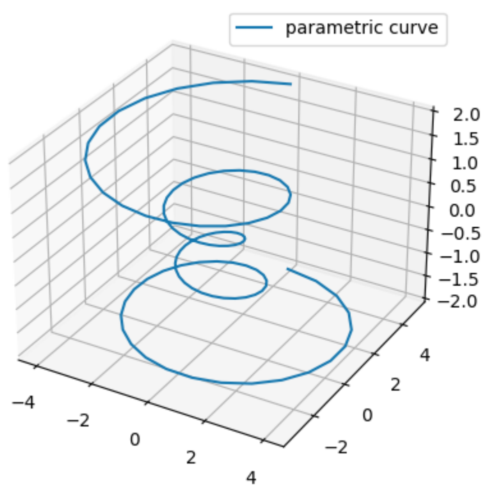

Fig 1. An example 3D trajectory plot.

Consider the initial state of *Ceres* to be $[10\ 10\ 10\ 0\ 0\ 0]^T$ in the CW frame, where the values are specified in *km*. Consider the ISS to be on a circular orbit of radius (semi-major axis) 6770 km around the Earth. The gravitational parameter of the Earth is 3.986 x 10^{14} m^3s^{-2}. Consider the maximum thrust acceleration capability of *Ceres* to be 0.07 ms^{-2} in all three dimensions. Since only the kinematics are considered, the mass or thrust magnitude of *Ceres* is not required. The Performance Index J is required to be minimized as much as possible; however, an upper bound is not provided for this project. Students are encouraged to compare their J value with other project groups and work on improving it. For this project, the docking can be considered achieved if *Ceres* approaches ISS at a distance of less than 10 m with a velocity magnitude of less than 0.5 ms^{-1}.

Please note that *Ceres* approaching the ISS with a high relative velocity at any time is dangerous to the safety of the ISS and should be avoided by design.

5. Milestones

The following milestones are required to be accomplished in the successful completion of this project. These milestones are intended to provide technical guidance and direction to the students for the completion of the project. Milestone 1 relates to the design application of theoretical concepts covered in lectures. Milestone 2 requires utilizing prerequisite course knowledge and the application of creativity and critical thinking. Milestone 3 may require some additional reading and knowledge acquisition.

Milestone 1A: Preliminary study of the motion of *Ceres* relative to the ISS.

The ability to model (predict) the trajectory of *Ceres* for a given initial state (i.e. initial position and velocity) in the CW frame is fundamental to this project task. This milestone requires the numerical propagation of the trajectory of *Ceres* using CW equations (Eq. 2), which represents the relative motion.

At this stage, any built-in software function that performs numerical propagation/solution can be used. A possible candidate is the Runge-Kutta propagator which has implementations in the libraries of commonly used languages such as Matlab and Python. Students may choose any initial state and acceleration profile (including no acceleration) for *Ceres*.

The milestone also requires a 3D graphical plot of the resulting trajectory (refer to Fig. 1) from propagation. Standard plotting functionality in programming languages can be used. The plotting is trivial once the output from the propagation is available.

This milestone forms part of the preliminary review in Week 6 (See Assessment Criteria in Section 7).

Milestone 1B: Modeling the trajectory of *Ceres* by implementing a numerical propagator.

This milestone requires the implementation of a numerical propagator in code, which is to be used for the remainder of the project. This replaces the built-in software function used for initial analysis in Milestone 1A. Background to numerical propagation will be covered in Week 2 lectures. Lab 1 in Week 4 will also provide additional hands-on experience on this topic.

Students may implement any type of numerical propagator. Students will need to use their propagator to model (propagate) the trajectory of *Ceres*. Students may choose any initial state and thrust acceleration profile (including no acceleration) for *Ceres*. As in Milestone 1A, a 3D graphical plot of the resulting trajectory is required.

This milestone also requires demonstrating the correctness of the implementation. The outcome from Milestone 1A serves as a reference to compare and verify the correctness of the trajectory modeling. It is advisable to make a comparison for different initial conditions. Note that any numerical propagator will carry estimation errors, so it is sufficient that the trajectories are reasonably similar for identical initial conditions. It is, however, essential to ensure that any major functional error does not get carried forward and invalidate the correctness of subsequent milestones.

<u>Milestone 2:</u> Implementation of a control scheme that enables *Ceres* to closely approach the ISS at an acceptable relative velocity and achieve docking.

From the implementation of Milestone 1, we now know (or can predict) the state (position and velocity) of *Ceres* at any given time. The target state of *Ceres* for successful docking is known from the definition of docking provided in Section 4. This milestone requires implementation of a control scheme to perform this docking process. The control scheme would apply thrust acceleration *u* (which is the control input) to progressively guide *Ceres* towards ISS. The control input would be reflected in Plot 1, requested in Section 4.

The control algorithm itself is not required to be implemented from scratch in code, and standard software functions can be used. Implementing a control scheme, however, may require equipping an algorithm with task specific parameters and often requires the tuning of coefficients as well, which may involve a trial-error process. Students are encouraged to utilize knowledge from pre-requisite courses to this end. While any control algorithm can be used for this purpose, not all of them may minimize the performance index *J*.

<u>Milestone 3:</u> Modeling of the measurement acquisition by *Ceres*.

In practice, the CW equations modelled in Milestone 1 may not predict the state of *Ceres* accurately, because of the effects of unmodeled accelerations and approximations in the dynamics. For example, the consideration of Earth as a point mass is not accurate, and its gravity has complex perturbing influences. Therefore, in the design of a practical system, the predicted (i.e. propagated) state of *Ceres* from Milestone 1 must be corrected using measurements/observations of the ISS obtained by sensors onboard *Ceres*. A filter can perform this task by comparing the measurements obtained from onboard sensors to the predicted measurements generated onboard based on the predicted state.

The completion of this milestone requires the modeling of the predicted measurements generated by *Ceres* during its operation. The measurements expected are the image coordinates of ISS obtained by an onboard camera and line-of-sight range obtained by an onboard ranging device. Consider a simple pin-hole camera model with appropriate parameters for focal length, field of view and initial orientation that obtains the x-pixel and y-pixel coordinates as measurements. Alternatively, an angles-only measurement of ISS can also be considered (azimuth and elevation). Consider the ISS-*Ceres* distance as the range measurement from the ranging device. Note that any practical sensor reading will carry measurement noise (sensor noise), and this must be included in the output (for e.g. in the form of an additive Gaussian noise to the measurement). Supportive information on this milestone will be provided in the discussion session in Week 2.

For simplicity and to reduce the scope of this project, it is not expected to correct the prediction from Milestone 1 using the measurements. This will require simulation of the true dynamics as well as the modeling and estimation of attitude (orientation) of *Ceres.* However, a simple pseudo-code/explanation of how this correction can be achieved using a filter is encouraged in the report.

6. Remarks

- Students can use any programming language convenient for them.
- The ISS can be represented as a point in 3D-space for this exercise, which in turn represents the target docking location. No 3D graphical representation of the ISS or its ports is required.
- For most parts, Milestones 2 and 3 can be implemented independently of each other, so students would not need to follow a sequential workflow.

7. Assessment criteria

As indicated in the class summary, the design project contributes 30% to the final grades. The grade distribution and rubric are provided below.

<u>Preliminary review:</u> (total 5%)

This is to ensure that students are prepared to undertake the design activity and to help alleviate project risks towards the end of the semester. Preliminary review includes the completion of Milestone 1A. This will be evaluated in Week 6 (see Section 9. Schedule).

- The group has a clear plan and task assignment among the members to complete the project by Week 12. The programming language/simulation environment and any other tools to be used for the design have been tentatively selected by the group. (1%)
- The CW equations with input acceleration are implemented/represented in a state-space form in code. (1%)
- Numerical propagation was performed using a built-in software function (2%)
- A 3D graphical plot representing the resulting trajectory is displayed. (1%)

Milestone review: (total 10%)

This provides an opportunity for the students to communicate technical solutions to design problems. Milestone review covers Milestones 1B and 2. This will be evaluated during the design project time in Week 9 (see Section 9. Schedule). Referring to their source code implementation for these Milestones, students are required to present their approach, factors that were taken into consideration, and how the solution was implemented. The evaluation is interactive, no slides are expected, and the code implementation itself is to be presented (on screen). This review assesses the implementation, and theoretical information is to be communicated in the project report.

- Demonstration of the completion of Milestone 1B. (2.5%)
  - This includes the implementation of the propagator.
  - Verification of the implementation's functional correctness.
- Demonstration of the completion of Milestone 2. (4.5%)
  - A control scheme was implemented in code for the problem and simulation with the control input was performed.
  - Any required tuning or optimization for the control scheme was performed and efforts were made to achieve docking.
  - Efforts to minimize the performance index *J.*
- Response to technical questions related to the project by the evaluator (at least one question for the group). (3%)

Report of the design project: (total 15%)

The design project report is required to be uploaded to *Wattle* at the completion of the project in Week 12. Milestone 3 is evaluated in the written report only. The report is expected to contain the following information.

- Introduction section that describes the theoretical background on space rendezvous, CW equations, and the problem statement. Students are encouraged to engage in additional reading to develop the introduction. (1%)
- Theoretical background to the numerical propagation method used for the design implementation. Please include all relevant equations. (1.5%)
- Theoretical background to the control scheme and approaches used for the design implementation. Please include all relevant equations. (2.5%)
- Clear graphs for Plots 1-4 mentioned in Section 4. (2%)
- Detailed 3D plot of the final approach trajectory as outlined in Section 4. (0.5%)
- Successful docking was achieved as evidenced by the plots, or adequate technical justification is provided as to why this could not be achieved. (2%)
- Completion of Milestone 3. Please describe the approach, how the modeling was performed, and any relevant equations. Please also include the measurements/plots. (3.5%)
- Further study: (2%)

  For this part, no modeling, simulations, or numerical analysis is required. A technical response of approximately half a page in response to the following question is expected in the report. Students may find Lecture 7 content useful to this end.

  In this design project, for simplicity, we did not take into consideration the attitude (orientation) modeling of Ceres. In practice, however, all actuators and sensors are mounted on the spacecraft bus. Describe how the consideration of attitude would alter or influence the overall design process followed in this project and what changes will need to be made to the design process or algorithms to achieve docking.

There is no minimum page/word count requirement for the report. All relevant technical information, however, must be adequately described. The report, excluding any appendices, shall not exceed 15 pages. Please attach the code to the appendix in text format (readable version). Alternatively, students can upload the code file with the report.

9. Schedule

The following table summarizes design project related activities from the class schedule. It is expected that the students will commence working on the project in Week 3.

Week 2	10am-12pm Lab 1.35	Discussion [required]
Week 4	10am-12pm Lab 1.35	Drop-in [optional]
Week 6	3pm-5 pm Lab 1.35	Drop-in [optional]

Week 6	9am-11 am Lab 1.35	Preliminary review [required]
Week 7	10am-12pm Lab 1.35	Drop-in [optional]
Week 9	10am-12pm Lab 1.35	Milestone review [required]

Please note that the project report is due at the end of Week 12.

10. Contact

Please email u1134790@anu.edu.au if you have any questions about the design project activity. Additional supportive reading material will be uploaded to the *Wattle* section for your review.

Document version	Remarks
1	Full draft.